习题7

* 1. **阶乘问题（循环）**

1.1.1.算法描述

阶乘问题，我们有两种方式，这边第一种是循环

1.1.2.伪代码

factorial

//对n进行求阶乘

//输入:需要求阶乘的数值n

//输出:阶乘的求值

val<-1

for i 1 to n do

val=val\*i

return val

1.1.3.算法实现

int factorial(int n){

int val=1;

for(int i=1;i<=n;i++) val\*=i;

return val;

}

* 1. **阶乘问题（递归）**

1.2.1.算法描述

阶乘问题的第二种求解方式就是递归

1.2.2.伪代码

factorial

//对n进行求阶乘

//输入:需要求阶乘的数值n

//输出:阶乘的求值

if n≠1 do

return n\*factorial(n-1)

end

1.2.3.算法实现

int factorial(int n){

if(n!=1) return n\*factorial(n-1);

}

* 1. **算法小结**

我们在求解阶乘时，循环，时间复杂度为O(n)，空间复杂度为O(1)，在递归求解的时候，我们时间复杂度为O(n^n)，空间复杂度为O(n)，当然我们还可以用数学进行推导阶乘的公式，那么我们求得的时间复杂度可以优化为O(1)。